Title	ネットワークの対称性と、k-LEADER(S)の計算能力(理論計算機科学とその周辺)
Author(s)	坂本, 直志
Citation	数理解析研究所講究録 (1992), 790: 194-200
Issue Date	1992-06
URL	http://hdl.handle.net/2433/82651
Right	
Type	Departmental Bulletin Paper
Textversion	publisher

数理解析研究所講究録
第 790 巻 1992 年 194-200

ネットワークの対称性と、 k-LEADER(S) の計算能力

坂本 直志
東京工業大学 理学部

概要

分散アルゴリズムとは、グラフ上の頂点に配置した各プロセッサに、辺上でメッセージをやりとりし合いながら同一のプログラムを実行させることによって問題を解決するアルゴリズムのことをいう。

分散アルゴリズムでは、プロセッサ番号を与えたりグラフサイズを教えるなど、ネットワークに関するある種の情報を初期条件として与えることがある。

本論文では、初期条件として単純に k 個のプロセッサを別の状態から動かすことを考える。この初期条件がネットワークの計算能力にどのような影響を与えるかを、決定性分散アルゴリズムによる還元可能性によって比較する。

1 はじめに

ネットワーク上の複数のコンピューターが、互いにメッセージを通信しながら、同じプログラムを実行し、何らかの問題を解くようなアルゴリズムを分散アルゴリズムという。ネットワークは、計算機と、構造を示すグラフにより指定され、このグラフの頂点に計算機があり、辺は通信線を表す。各頂点上の計算機は、自分に接続している辺の数 k が与えられていて、それらの辺を 1 以上 k 以下の連続した番号で区別することができる。この番号を、その頂点におけるその辺の port number と呼ぶ。各計算機は、この port number を指定して隣接している特定の頂点へメッセージを送ることができる。また受け手の計算機は、このようにして送られたメッセージが自分に接続しているどの頂点から送られてきたものかを、辺の port number により知ることができる。

分散アルゴリズムの問題に、ある情報の放送、故障したプロセッサの発見、分散環境の OS における複数のプロセスや資源の管理などがある。これらのアルゴリズムの設計で、ネットワークの構造や、ネットワークに存在する計算機の数などのネットワークに関するある種の情報を、初期条件として与えることが行なわれてきた。また、いくつかの研究では、異なる初期条件のもとでアルゴリズムが設計されている。これらの中で主なものは、 [坂本 91] において、比較分類が行なわれている。

本研究では、 [坂本 91] で完全性の示された LEADER という 1 つの頂点のみ別の状態から計算を始める初期条件を拡張した、 k-LEADER(S) という初期条件の計算能力

を、初期条件 A から分散アルゴリズムにより初期条件 B が生成できるかどうかという還元可能性の手法で調べた。

本研究で比較した初期条件は次の 6 つとその組合せである。

1. EC(Edge Coloring)
   どの頂点に対しても、それに接続している辺に付けられた番号全てが異なるような辺への番号の付け方を 1 つ与える。この番号は、必ずしも、全ての辺に異なる番号がつく必要はなく、また、全体として連続した番号である必要もない。

   各頂点には、辺の port number とその辺の番号の対応表の形でこの番号付けが与えられる。

2. C(Coloring)
   どの頂点に対しても、それに隣接した頂点同志は異なるような頂点への番号の付け方を 1 つ与える。これも、全ての頂点に異なる番号が付く必要はなく、また、全体として連続した番号である必要もない。

   各頂点には、その頂点の番号が与えられる。

3. SZ(Size)
   各頂点にグラフサイズを与える。

4. ST(Structure)
   各頂点に、グラフ全体の構造と port number のリストを示す情報を与える。但し、自分がどの頂点であるかという情報は含まない。

5. k-LEADER(S) (k-Leader(s))
   グラフサイズが $2k$ 個より大きいならば、k 個の頂点にだけ $\langle k, 1\rangle$ を与え、他の頂点全てに $\langle k, 2\rangle$ を与える。

6. EE(Edge electing)
   1 つの辺の両端の頂点に、その辺を示す port number を与え、他の頂点には、0 を与える。

また形式上、分散アルゴリズムで最低限必要な初期条件として次のようなものを考える。

7. $\emptyset$
   各頂点に、次数を与える。

2 準備

ネットワークの頂点上での計算のモデルとしては，決定性 Turing machine を使用する. ただし， RAM モデル等を使用しても本論文の結果はそのまま成り立つ.

$a|b$ で a は b を割り切ることを表し、$a \nmid b$ で、その否定を表す。$\langle a, b\rangle$ を順序対、または pairing function と混用する。

本研究で考えるグラフはすべて有限無向単純グラフである。グラフ $G=(V,E)$ について、$|V|=N$ とする。$v \in V$ について $\deg(v)$ を頂点 v の次数とする。

network を (G, F_G, M) という 3 つ組で表す。$G=(V,E)$ はグラフ、$F_G=\{f_v \mid v \in V\}$、但し、f_v は、$\{1,\ldots,\deg(v)\}$ から v に隣接している辺の集合への上への 1 対 1 の写像とする。$f_v(i)=e$ のとき、i を v における e の port number と呼ぶ。

M は 決定性 Turing machine で、作業用テープのほかに、入力テープ、出力テープ、送信用テープ、受信用テープをそれぞれ 1 本ずつ持っている。頂点 v 上に配置された Turing machine を processor と呼ぶ。各 processor は、本論文では、同期して動くものとする。

各 processor がメッセージを送るには、送りたい辺の port number を送りたいメッセージに付け、送信用テープに書き込み、送信状態に入ることにより送られる。そして、メッセージを送られた側の受信用テープにメッセージを送った頂点の受け側から見た port number がメッセージに結合されて書き込まれる。もし、同時に 2 つ以上のメッセージが届いた時は、若い port number がついている辺から送られてきたメッセージが先に読まれるものとする。

情報の列、$\mathbf{x}=\langle x_1,\ldots,x_N\rangle$, $\mathbf{y}=\langle y_1,\ldots,y_N\rangle$ について、$(G,F_G,M)(\mathbf{x})=\mathbf{y}$ とは、各 processor $p_1,\ldots,p_N$ に $x_1,\ldots,x_N$ を各々入力し、アルゴリズムを実行した後の出力が $y_1,\ldots,y_N$ であることをいう。1 つでも出力を出さない processor が存在する時、$(G,F_G,M)(\mathbf{x})$ は未定義とする。

初期条件 A, B と決定性分散アルゴリズム M について、任意のグラフ G、その port number 関数の任意の集合 F_G、任意の順序対 $\mathbf{x}$ に対し、$\mathbf{x}$ が G において B の正しい初期条件であれば、$(G,F_G,M)(\mathbf{x})$ は必ず停止し、G における A の正しい初期条件である時、$A \le B$ via M と表す。

また、$A \le B \Leftrightarrow (\exists M)[A \le B \text{ via } M]$ と定義する。

$<, \equiv$ を次の様に定義する。

$$
\begin{aligned}
A \equiv B &\Leftrightarrow A \le B \wedge B \le A \\
A < B &\Leftrightarrow A \le B \wedge B \not\le A
\end{aligned}
$$

命題 2.1 $\equiv$ は、同値関係であり、同値類を作る。

$\le$ は、$\equiv$ の作る同値類に関してそれぞれ反射律、反対称律、推移律を満たすので、同値類に関して、半順序関係の性質を持つ。

初期条件 A が $\le$- 完全であるとは、同型なグラフを全て同一視すると、全ての初期条件 B について、$B \le A$ を満たすことである。

3 主要結果

3.1 還元性の作る lattice について

$\leq$ は、同値類に関して半順序関係になっており、同値類を表すのにその代表元を用いることにすると、上界として complete 、下界として $\emptyset$をもつので、lattice を作る。

初期条件 A, B について、 $\mathbf{x}$ が G において初期条件 $A \sqcup B$ を満たすとは、$\mathbf{x} = \langle\langle y_1, z_1\rangle, \ldots, \langle y_N, z_N\rangle\rangle$ について、$\mathbf{y} = \langle y_1, \ldots, y_N\rangle$ が G において初期条件 A を満たし、 $\mathbf{z} = \langle z_1, \ldots, z_N\rangle$ が G において初期条件 B を満たすことである。

閉区間 $[Q, R] = \{X \mid Q \leq X \leq R\}$ と $S \sqcup [Q, R] = \{S \sqcup X \mid X \in [Q, R]\}$ を定義する。

以下の lemma は、一般の lattice について成り立つ。

[補題 **3.1**] $[A, B], [C, D]$ について、$(\forall x \in [A, B])\ (\forall y \in [C, D])\ [x \not\leq y] \Leftrightarrow A \not\leq D$

[補題 **3.2**] $[A, B]$ と c について、$c \sqcup A \not\leq B \Rightarrow (\forall x, y \in [A, B])[c \sqcup x \not\leq y]$

3.2 k-LEADER の作る階層構造

[定理 **3.3**] 1-LEADER は $\leq$- 完全。[坂本 91]

以下、 1-LEADER の作る同値類のクラスを complete と表すことがある。

[定理 **3.4**] $(\forall x, y \in [\emptyset, \mathrm{ST} \sqcup \mathrm{C}])\ [k\text{-LEADER(S)} \sqcup x \not\leq y]$

[定理 **3.5**] $(\forall x \in [\emptyset, \mathrm{C}])\ [\mathrm{SZ} \sqcup x \leq k\text{-LEADER(S)} \sqcup x]$

[補題 **3.6**] k 個の leader が存在する時、その leader の張る生成木により、そのネットワークは k の約数個の互いに同型な構造を持つ生成木に分割することができる。

隣接した生成木を統合していく統合化アルゴリズムを用いるが証明は割愛する。

(定理 3.5 の証明)　$\mathrm{SZ} \leq k\text{-LEADER(S)}$ を示す。

補題 3.6 で隣接した生成木を統合する際に、統合した木の本数を覚えておくことにより、停止する際に (生成木のサイズ)$\times k \div$ (統合した木の数) の値を出力することとすると、この値はグラフサイズになっている。 □

[定理 **3.7**] $(\forall x \in [\emptyset, \mathrm{C}])\ (\forall y \in [\emptyset, \mathrm{EC}])\ [\mathrm{ST} \sqcup x \not\leq \mathrm{EE} \sqcup y]$

[定理 **3.8**] $k \geq 2 \Rightarrow (\forall x, y \in [\emptyset, k\text{-LEADER(S)} \sqcup \mathrm{C}])\ [\mathrm{ST} \sqcup x \not\leq y]$

[定理 **3.9**] $(\forall x \in [\emptyset, \mathrm{C}])\ [k | l \Leftrightarrow l\text{-LEADER(S)} \sqcup x \leq k\text{-LEADER(S)} \sqcup x]$

[定理 **3.10**] $k \geq 2 \Rightarrow (\forall x \in [\emptyset, \mathrm{C}])\ [\mathrm{EE} \sqcup x \not\leq k\text{-LEADER(S)} \sqcup x]$

[定理 3.11] $(\forall x \in [\emptyset, \mathrm{EC}])\ [2|\,k \Leftrightarrow k\text{-LEADER(S)} \sqcup x \leq \mathrm{EE} \sqcup x]$

[定理 3.12] $k \geq 3 \Rightarrow (\forall x \in [\emptyset, k\text{-LEADER(S)} \sqcup \mathrm{ST}])\ [x \sqcup \mathrm{EC} \not\leq x]$

[定理 3.13] $\mathrm{EC} \leq 2\text{-LEADER(S)}$

[定理 3.14] $\mathrm{EE} \sqcup \mathrm{C}$ は complete.

(証明) 選ばれた辺の両端の頂点の着色が異なれば、どちらか一方の頂点を区別できるので、1 つの leader を決めることができる。 □

[定理 3.15] $2 \nmid k \Rightarrow (\forall x \in [\emptyset, \mathrm{ST}])\ [k\text{-LEADER(S)} \sqcup \mathrm{C} \sqcup x \leq k\text{-LEADER(S)} \sqcup \mathrm{EC} \sqcup x]$

(証明) $k\text{-LEADER(S)} \sqcup \mathrm{C} \leq k\text{-LEADER(S)} \sqcup \mathrm{EC}$ を示す。

統合化アルゴリズムを用いる際に、大小比較の条件に各頂点の情報として、相対的な頂点番号のほかに、接続している各辺について、 port number, その辺の色、隣接している頂点の相対的な番号とその辺の port number を用いる。

すると、統合化アルゴリズムが終了した時点で、同じ番号の振られた頂点同志は、隣接しない。

同じ番号 n の振られた頂点同志が隣接していると仮定すると、同じ番号の隣接の仕方は、

1. 隣接している辺の port number が等しい場合
2. 隣接している辺の port number が異なる場合

に分類できる。

統合化アルゴリズムが終了しているので、k の約数個の n と振られた頂点全てがこのどちらか一方だけになっている。1. の場合は、 k の約数は奇数なので、これは矛盾である。

2. の場合、異なる port number を a, b とすると、比較の条件から、同じ port number の辺は、同じ着色になっているので、 port number a, b の辺だけを着目すると、番号 n の頂点だけが次数 2 のグラフを作っていて、しかも、辺を 2 色で塗ることができる。(全次数が 2 なので、サイクルの集まりになっている。)

しかし、 補題 3.16 より、 2 色で塗ることのできない奇数長のサイクルが存在するので、矛盾である。

つまり、統合化アルゴリズムが終了した時点では、同じ相対的な番号が、互いに隣接しないので、これは頂点への彩色の条件を満たしている。 □

[補題 3.16] 奇数個の頂点からなる次数の等しいグラフで、偶数長のサイクルのみを持つものは存在しない。但し、孤立点は長さ 1、頂点同志を結んだただ 1 本の辺は長さ 2 とする。

(証明) 偶数長のサイクルのみを持つグラフは、頂点を 2 彩色可能である。よって、このグラフは 2 部グラフになるが、頂点数が奇数なので、頂点を等しい数に分割できない。

頂点の分割を a, b $(a < b)$ 、全ての頂点の次数を d とすると、分割 a から出る辺 ad 本と、分割 b から出る辺 bd 本は等しくなく、2 部グラフを構成できず矛盾である。 □

[定理 **3.17**] $2|\,k \Rightarrow (\forall x \in [\emptyset, k\text{-LEADER(S)} \sqcup \text{ST} \sqcup \text{EC}])\ [x < x \sqcup \text{C}]$

[定理 **3.18**] $k \geq 2 \Rightarrow k\text{-LEADER(S)} \sqcup \text{ST} \sqcup \text{C}$ は、 complete でない。

3.3 まとめ

今まで示した結果をまとめると次のようになる。

[系 **3.19**] k-LEADER(S) 内の構造は、図 1 の通りである。明らかでない関係は、全て incomparable である。($A \rightarrow B$ は $A < B$ を意味する。)

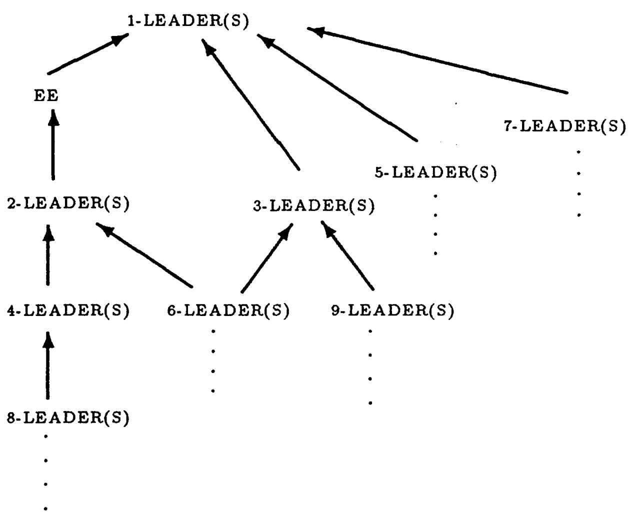

図 1:

[系 **3.20**] k-LEADER(S) と、他の初期条件との関係は、図 2 の通りである。

参考文献

[Ang80] D. Angluin. Local and global properties in networks of processors. *Proc. 12th ACM Symp. on Theory of Computing*, pp. 82–93, 1980.

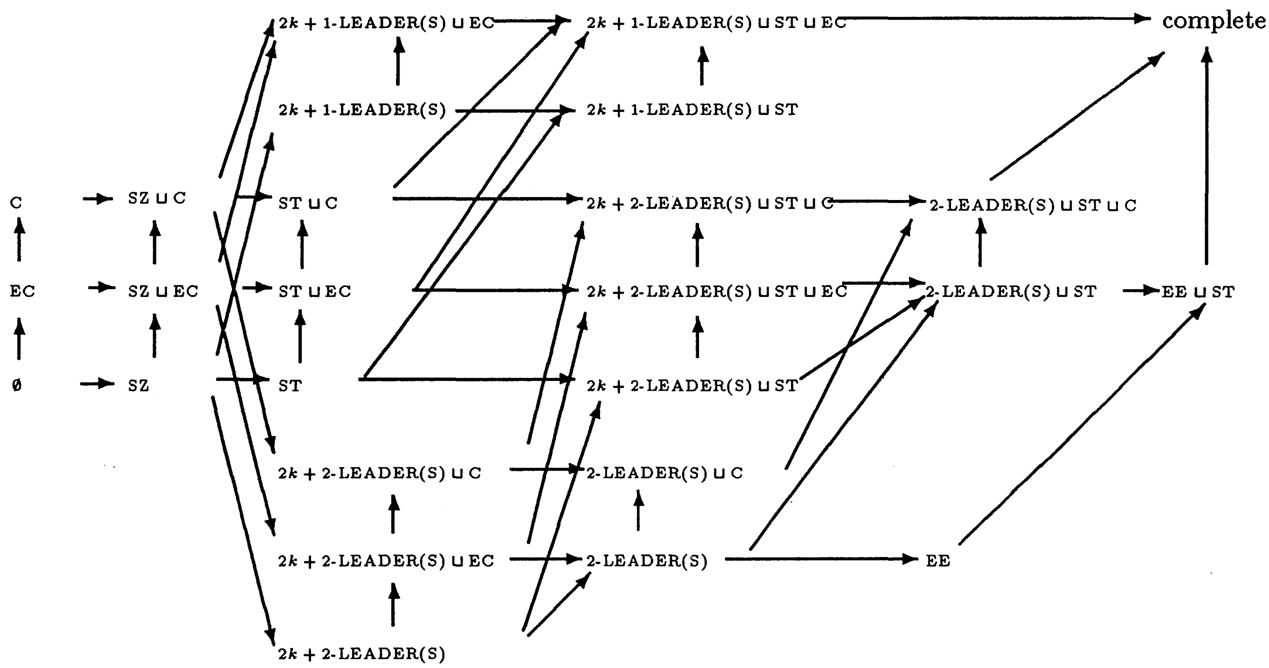

図 2:

[FPRU90] U. Feige, D. Peleg, P. Raghaven, and E. Upfal. Randomized broadcast in network. *Lecture Notes in Computer Science*, Vol. 450, pp. 128–137, 1990. SIGAL'90.

[GHS87] R. G. Gallager, P. A. Humblet, and P. M. Spira. A distributed algorithm for minimum-weight spanning trees. *ACM Transactions on Programming Languages and Systems*, Vol. 5, No. 1, pp. 66–77, January 1987.

[IR81] A. Itai and M. Rodeh. Symmetry breaking in distributive networks. *Proc. 22nd IEEE Symp. on the Foundations of Computer Science*, pp. 150–158, 1981.

[YK88] M. Yamashita and T. Kameda. Computing on anonymous networks. *Proc. 7th Ann ACM Symp. on Principles of Distributed Computing*, pp. 117–130, 1988.

[YK89] M. Yamashita and T. Kameda. Electing a leader when processor identity numbers are not distinct. *Lecture Notes in Computer Science*, Vol. 392, pp. 303–314, 1989. Distributed Algorithms.

[坂本 91] 坂本直志. 分散アルゴリズムの初期条件の相対的複雑さについて. 信学技法, Vol. 91, No. 369, 1991. COMP 91-81.